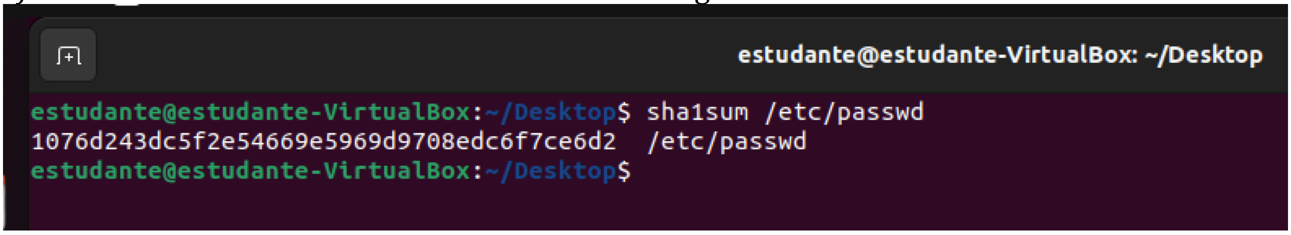
1st Hands-on Assignment

Comparing every file on a system with their original installed versions is an essential operation to guarantee that the software which exists on a computer was not altered. Instead of comparing the content of files directly, we can compare the hash of the files, which can be thought of as a summary of their contents. The hash (or summary) of a file is obtained using a function *F* on a series of bytes *S* of a file, which produces a sequence of bytes *H* which is always of a fixed length. *F* has the property that, knowing *S* and *H*, it's very difficult to obtain a sequence S´ such that F(S') == H. An example of the function *F* is *sha1*; visit Wikipedia to learn more about the *sha-1* hash function.

The Linux command *sha1sum <filename>* calculates a hash of 20 bytes (160 bits) of the file <*filename*>. The followin screenshot shows the usage of the *sha1sum* command.



```
estudante@estudante-VirtualBox:~/Desktop$ sha1sum /etc/passwd
1076d243dc5f2e54669e5969d9708edc6f7ce6d2  /etc/passwd
estudante@estudante-VirtualBox:~/Desktop$
```

Note that the command writes a line of text with the hash and name of a file, separated by two spaces.

This hands-on assignment involves writing a *bash* script which verifies the hashes of files contained in a directory with a text file with the genuine, expected hashes.

The script should receive two arguments.:

- The name of a directory *dir*, containing various subdirectories with files inside; and
- The name of a text file which contains a line for each file contained in the drectory *dir* in the format produced by the *sha1sum* command.

The script should write the following to the terminal:

- OK if all of the file hashes in the directory pointed to by the first argument are equal to those which appear in the file pointed two by the second argument; or
- The line number(s) and filename(s) for which there were differences.

In order for the implementation to be valid, it must follow the following instructions:

1. The *bash* script must use the *find* command to produce a text file in which each line contains an output produced by the *sha1sum* command.
2. The comparison between the hashes of files contained in the directory and the genuine hashes must use the *compare* program written in the 1st part of the class.
3. The file which is passed as the 2nd argument is sorted by filename. This means that the file produced in point 1 must also be ordered by filename. To perform this ordering, please use the *sort* command. Assume that the two files have the same number of lines.

In CLIP, under the section *Documentação de Apoio -> Problemas*, there are two files: the archive *teste.tar* with directories and files, and the file *hashs_corretos* that contains the hash of the files in *teste.tar*. For your experiences do the following:

- Download the files *teste.tar* and *hashs_corretos* into your working directory.
- Expand the tar file *teste.tar* with the following command: *tar xvf teste.tar*
- Invoke your script with the arguments *teste* and *hash_corretos*. No different hash should be detected.
- Try now to modify some files in one of the subdirectories of *teste* and execute again your script to verify which files were modified and for that reason have different hashes than the ones in the file *hash_corretos*.

How to submit: to be announced soon through a message sent via CLIP.